AN INTELLIGENT MOBILE APPLICATION CAPABLE IN HOMELESSNESS INTERVENTION BY UTILIZING SOCIAL NETWORKING AND AN AI CHAT BOT

Yuxi Sun[1], Victor Phan[2]

[1]Tarbut V'Torah, 5 Federation Way, Irvine, CA, 92603
[2]Computer Science Department, California State Polytechnic University, Pomona, CA 91768

ABSTRACT

This project is an application made with the purpose of homelessness intervention [1]. It allows for the homeless to be able to send requests to a server, and the volunteers will help fulfill them to help the homeless get their life back together. The three major components of this are the volunteer app, kiosk app, and the AI chatbot [2]. The volunteer app is the app used by the volunteers to fulfill the requests sent by the kiosk. The kiosk app is what sends the requests to the server. The AI chatbot is a part of the kiosk app that allows the user to input questions for the bot to respond to. An experiment that was conducted was testing out the accuracy of the AI chatbot, with the result being 83%. Utilizing an app is a good idea because of how prominent cell phones are. This makes it very easy to receive and fulfill the requests as soon as possible.

KEYWORDS

Homelessness, Volunteer, AI, social network

1. INTRODUCTION

The homelessness epidemic has been a worldwide issue that governments are still today trying to solve to no avail [3]. Homelessness is a varied and complex issue to tackle, as it is caused by a myriad of socio-economic issues working in tandem. Cost of living and health care has been very high. Natural disasters can result in a person never being able to fully recover from the damage [4]. That and the lower incomes that are received will make it very hard to get off the streets. The byproduct of these socio-economic issues is the displacement of thousands of citizens onto the streets [5]. This issue is growing year by year and many solutions have been implemented to stop it. Attempts have been made for affordable housing, as well as lots of nonprofit organizations and government support made to try and prevent this issue. Unfortunately, homelessness has yet to decrease after these solutions have been tried. While they may help prevent some from becoming homeless, the amount of people on the streets have not gone down. Homelessness affects all corners of society negatively. The streets of cities are filled with the homeless and it's only a matter of time before there's no space left for them to take. Average citizens unfortunately are not positive towards the homeless. Instead of trying to help, they tend to avoid the homeless. In summary, homelessness is a pervasive issue that affects nearly everybody in modern day society. This is a major issue that cannot be solved by simple policy changes.

There have been other methods to try and solve the constantly increasing problem that is homelessness. The idea of using blockchain to better record the identity of the homeless to give them the correct treatment is one method that has been proposed. Then, there are other things that require the support of the government such as Critical Time Intervention and managing costs for basic needs such as rent to make the costs more manageable [6]. While the solutions presented are logical and very possible, there are still major shortcomings that may come from them. Whether it be due to the cost or heavy reliance on the government to fix it, they are the setbacks that make the proposals lose their theoretical effectiveness. This app does not have any of the major problems that were a setback for the other solutions. It is free and community based, so measures can be taken care of immediately.

This application will be able to give opportunities to the homeless in their time of need by providing them references to helpful government resources or helping them reach out to volunteers to request necessities such as food and shelter. By typing their request into the kiosk app, they are able to send this information to other users who are willing to volunteer and help [7]. Volunteers can see these requests by downloading a separate version and signing up for a free account. A kiosk version would be stationed near helpful volunteering resources such as food banks to encourage the homeless to seek these shelters and use the application to request further amenities in addition to what food banks and shelters already provide. Using the kiosk application, the person can either talk to the AI chat bot, or they can send a request [8]. The AI chat bot can give an immediate response to what the person is asking [9]. The chat bot will be able to understand natural language suggestions such as depression, a request for shelter, career counseling, or job assistance. If the chat bot is unable to help, sending a request will. This feature relies on human volunteers and they will be able to help where the AI chatbot could not. For instance, homeless people may request food or clothing through this feature. Homeless people have always been stigmatized as someone to avoid and look down upon. By using this app, homeless people can now be connected to volunteers directly, and this will lessen the perceived social gap between the homeless and those who can afford to live in the modern age.

One experiment conducted was to test the accuracy of how the AI Chatbot responds to what was being asked of it. It was asked questions and expected to be able to respond to the pre-designated category the question was in. This resulted in an accuracy of around 83%. The questions the bot was not able to answer were "I'm in debt" and "can I still get a job despite being homeless?". The conclusion was that the bot is not able to detect questions that were either too vague and can easily be put in multiple categories, or they were too complex for the response to be from just one category.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Developing the AI

The AI that is implemented into the chat system must be flexible and able to understand nuances in human speech. This, however, is difficult, so a concern with this is how exactly this should be implemented. To alleviate this, we will choose IBM Watson's chatbot service, which already has a strong backend for natural language processing [10]. We also intend for the AI to be as accurate as possible. This is easier said than done, and it does bring up the question as to how that accuracy is achieved. We provide the AI with a lot of training data with as many things related to the problem we are trying to solve as possible. Lastly, the AI is programmed to respond to everything related to homelessness. One concern is to figure out how exactly its behavior could

be helpful. To ensure quality service, the AI will provide relevant links to the user when asked. This makes it intuitive to interact with and convenient to use.

2.2. How Exactly Kiosks will Function

The intended use mechanism for this application is through a kiosk terminal. The operative question is thus how exactly these kiosks will function. The kiosk will come loaded with a specialized version of the app that will make it easy to use on a kiosk. This kiosk application is only one part of the proposed system of apps. One concern is how it would be connected to other apps such as a separate volunteer-oriented app. To connect the two, requests sent from the kiosk app are sent to a remote server, and the volunteer app will read information from this server. We believe that the kiosk system is a good implementation of this idea. There might, however, be potentially better implementations than a kiosk. Ultimately, we decided to use a kiosk system because it would be the most accessible and noticeable.

2.3. The App must Function Properly and Efficiently

The app on both fronts must function properly and efficiently. Otherwise, the user experience is inconvenient and users will not want this service after one use. To make the app experience satisfactory, the user interface must be simplistic, easily recognizable, and difficult to get lost in. In terms of accessibility, ideally anyone should be able to use this kiosk application, including the disabled and the impaired. One way to accomplish this is to make the user interface minimal. This includes, for instance, minimizing the amount of words on screen, and making them large, and in an easily readable font that can be read at a distance.

3. SOLUTION

The main structure of the app is an input for requests and having an output to where the requests get fulfilled. It is primarily made on Flutterflow, with the addition of the chatbot created on IBM Watson, and the server used to take in and collect the requests is Firebase. There are three main components necessary to make this structure work. First would be the kiosk app. This is the point of contact for the homeless, used to connect them to others more easily by allowing them to send a request and letting the volunteer app users or chatbot be able to see and respond to it. The volunteer app is the crucial component to make sure the requests get fulfilled. It is what allows the volunteers to be able to see the requests and help out. The chatbot is a main component for those who need an instant, reliable response. Depending on what's being asked, it can give links and other online resources to help out as best as it can. Each component is very essential to making this system work. A homeless person interacts with the app starting with the kiosk. They have three options: links to resources, discussing with a chatbot, and making a custom request. When they interact with the chatbot, the chatbot is able to recognize the user's topic of interest and provide external resources in response. When a user makes a custom request, they can provide their name, location, and requested items. This request is uploaded to the Firebase server and displayed on the volunteer app for people to interact with.
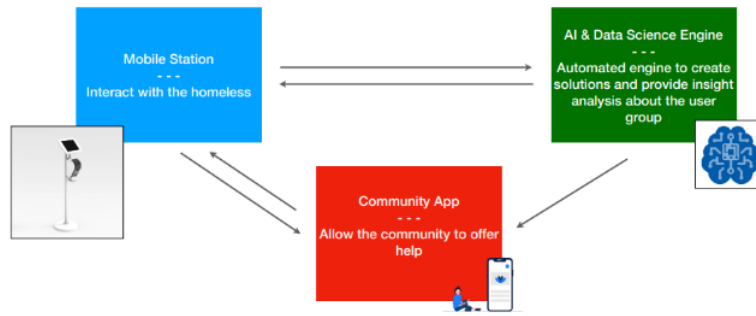
Figure 1. Overview of the solution

The kiosk app is meant for the homeless and the component that sends the requests to the server. By connecting Flutterflow to Firebase, it is able to collect the requests sent by the form and send it to the volunteer app. The kiosk app is also what contains the AI chatbot and a few already given resources, to give them the most immediate help.
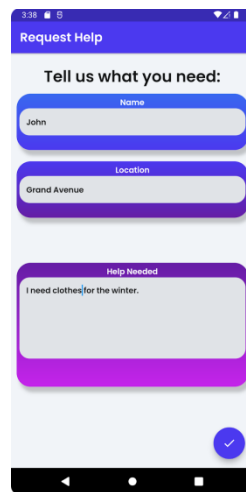

Figure 2. The screenshot of the AI chatbot

Figure 3. Screenshot of code 1

The first part of the code is validation. This makes sure the request form is actually filled in and prevents code injections. The second part is document creation to send to the server. The server being used here is Firebase, a Google provided service [11]. The document being created here is part of a collection that is stored using the Firestore sub-service. The document itself contains

multiple fields, and a field has a label and value. The labels here are the name, location, and request, while the value is the corresponding input depending on what was filled out in the form. These are based on text fields in the form page. The name is an identifier. It does not need to be the legal name. The location is similar in that it needs to be an identifiable location to meet up at. The request portion of the form would consist of specific directions on how to meet the required need or service. The document needs to be uploaded to the server to make sure the request is able to be sent to the volunteer app and seen by someone else. The last part of the code would be Navigating back. As soon as the request is sent, the app navigates back to the home page. This is to avoid accidental duplicate form submissions. If a duplicate form were to be submitted, then multiple volunteers would be assisting the same person, or a volunteer may try to accomplish a request that has already been fulfilled.

The purpose of the volunteer app is to collect the requests sent from the kiosk app and allow the volunteers to accept and take action. Taking the requests from Firebase, the volunteer app displays them and once they are accepted, the requests are put on a to-do list until they are fulfilled.
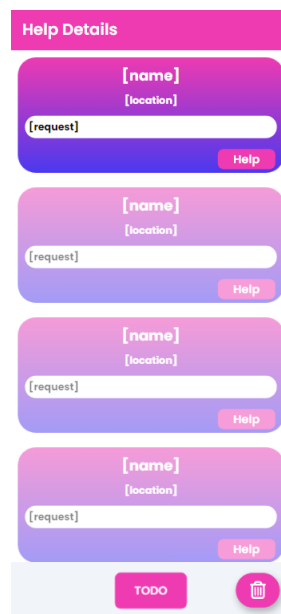

Figure 4. The screenshot of help page

```
// Generated code for this Button Widget...
Align(
  alignment: AlignmentDirectional(0.95, 0),
  child: Padding(
    padding: EdgeInsetsDirectional.fromSTEB(5, 5, 5, 5),
    child: FFButtonWidget(
      onPressed: () async {
        final todoCreateData = createTodoRecordData(
          helper: currentUserUid,
          name: listViewRequestsRecord.name,
          location: listViewRequestsRecord.location,
          request: listViewRequestsRecord.request,
        );
        await TodoRecord.collection.doc().set(todoCreateData);
        await listViewRequestsRecord.reference.delete();
        setState(() => _model.firestoreRequestCompleter = null);
        await _model.waitForFirestoreRequestCompleted();
      },
      text: 'Help',
      options: FFButtonOptions(
        width: 80,
        padding: EdgeInsetsDirectional.fromSTEB(0, 0, 0, 0),
        iconPadding: EdgeInsetsDirectional.fromSTEB(0, 0, 0, 0),
        color: FlutterFlowTheme.of(context).primary,
        textStyle: FlutterFlowTheme.of(context).titleSmall.override(
              fontFamily: 'Poppins',
              color: Colors.white,
            ),
        elevation: 2,
        borderSide: BorderSide(
          color: Colors.transparent,
          width: 1,
        ),
        borderRadius: BorderRadius.circular(8),
      ),
    ),
  ),
)
```

Figure 5. Screenshot of code 2

This code runs as soon as the volunteer accepts the request. A new document with the exact same fields as the original request is created, only now there is an additional field listing the UID of the helper and it is put in the "to-do" collection. This collection represents the set of requests that have been accepted, with the requests collection containing the unaccepted requests. Now that the new document has been created, the original request in the request collection will be deleted. The document has effectively been transferred from one collection to another, the only difference being that the newly created document contains the UID of the helper. Then, we refresh the database request and show that the volunteer's acceptance request has been approved. Refreshing the database request also makes sure new requests are shown for the user. That way, a request wouldn't be accepted multiple times, but still making sure the request is completed.

The AI chatbot is used for when the user is in need of an instantaneous online resource. It is programmed to respond to one of multiple topics. These topics are determined by the AI autonomously using natural language processing without having the end user explicitly state what it should respond with.

Figure 6. Screenshot of the respond

Figure 7. Screenshot of code 3

The code above will execute whenever a user types a message and hits the send button. The first thing that needs to be addressed are the message lists: the user message list, the bot message list, and the all message list. The user message list is the list of messages the user has sent. The bot message list are the responses the chatbot/server returns. The all message list is the list of both the user and the bot's messages in chronological order. Action 1 and 2 is about updating the app state when the message is sent. The app state is the background data handled by the application. It adds the user's message to the user messages list and the all message list. When the user sends a question, it is sent to the server. The server gives the message to the chatbot, and the server relays the chatbot's response back to the application. If it succeeds in giving a proper response, the response will be added to the bot message list and all message list and scroll to the bottom.

However, if it fails in giving a suitable response, an error response is added to the bot and all message lists.

4. EXPERIMENT

This experiment is testing the capabilities of the AI Chatbot in the application. To test this, it is given several different statements to test whether or not it will detect them correctly.

The goal is to test the AI Chatbot's accuracy in responding to what it's being asked. The main categories it's able to detect are jobs, education, housing, and mental health support. Anything that it finds that is not related to the 4 is not able to be given an insightful response. The data is then collected stating the AI's accuracy. Examples of what's being asked of the Chatbot are as such: "I am out of money and may go in debt soon", "where can I find a place to live?", "I want to continue my education", and "What can I do to get my life back together?"

	Question	Intended Topic	Response	Match?
1	Where can I go find a place to live?	Housing	Housing	Y
2	I want to continue my education.	Education	Education	Y
3	I'm depressed and in need of comfort.	Mental Health Support	Mental Health Support	Y
4	I'm out of money and may go in debt soon.	Job	Job	Y
5	What can I do to get my life back together?	Mental Health Support	Mental Health Support	Y
6	Housing near me.	Housing	Housing	Y
7	I need to get my degree again.	Education	Education	Y
8	I need a job.	Job	Job	Y
9	I don't have a house.	Housing	Housing	Y
10	I'm in debt.	Job	Confusion	N
11	How do I go back to college with no money?	Education	Education	Y
12	Can I still get a job despite being homeless?	Job	Housing	N
				83.33333333

Figure 8. Graph of experiment 1

The total accuracy of this experiment was around 83%. The questions the bot detected incorrectly were "I'm in debt." and "Can I still get a job despite being homeless?". "I'm in debt" is a vague phrase and may be misconstrued as a request for Housing services. The second question "can I still get a job despite being homeless?" is a complex question that is difficult for a bot to respond to. The question also seems to have been miscategorised because the bot detected "housing" as the more important category this question is asking about. The bot was likely trained more on detecting housing keywords than job keywords. Both categories are part of the intended category of "job", making the overall accuracy of this group of questions 50%. This is most likely due to the questions being too complicated for the bot to properly respond to and will incorrectly detect the intended category.

5. RELATED WORK

Anjum Kurshid and Ashish Gadnis propose the idea of using blockchain to record their identity digitally for doctors to be able to easily use to prevent misdiagnosis and unnecessary screenings [12]. These two methods are very different, with one being an identification and background check and the other sending requests. This app aims for a more general audience and is not intended specifically for medical purposes. Food, water, and other amenities may be requested in addition to medical needs. Blockchain is also very expensive to keep up. The app has to constantly be maintained on a server and it is expensive to maintain. Our app requires little maintenance in comparison.

Kaitlin Phillips of Salve Regina University proposes a policy reform for the US, which focuses on eliminating homelessness through community efforts, like Assertive Community Treatment and Critical Time Intervention [13]. Both policies are rehabilitative-centered treatments. The Danish government has used these policies for a long time. They are considered effective policies that the US has yet to incorporate, and the estimated cost of implementation is said to not cost more than current US intervention methods. Our application, by contrast, is a more tech-centric solution and our application is more focused on general assistance rather than rehabilitation. Both solutions, however, are centered around community assistance.

Another method proposed would be to reduce the cost of health and social services to make paying for houses easier [14]. This is effective as the cost for those kinds of support is very expensive and lowering them even a relatively small amount would be very beneficial. The results show a very low rate of interest to the point of just somewhere around one, resulting in the possibility of breaking-even and returning profit. The downside of this, of course, would be that it requires the change of a policy. Not only would there be a chance of this proposal being rejected, but it is also a potentially long process, leaving nothing done within that time. Being just an application, it would have no need for government support and, therefore, be able to take action immediately.

6. CONCLUSIONS

The general limitations to this app are that it is very dependent on the help of volunteers. Policies need to be made to make sure that no one is abusing the app and to make sure both sides are safe. The chatbot is also very limited in its response. It will only give an answer to a question it recognises as one of the categories it was given to detect. ChatGPT would be a good way to improve the app by giving more varied and unique answers to the questions asked [15]. Because this app is open for basically anyone to use it, there isn't much stopping non-homeless people from abusing the functions of the app. Therefore, policies must be made to prevent that from happening. There should also be policies made to secure the safety of both the volunteer and the homeless when fulfilling a request.

While homelessness will continue to persist, there should be more active effort to try and get more people off the streets. Trying to prevent homelessness in the first place will be very difficult as the causes are innumerable and sometimes just out of everyone's control. That is why this app focuses on stopping the homeless from getting to a point of no return.

REFERENCES

[1]   Ploeg, Jenny, et al. "A case study of a Canadian homelessness intervention programme for elderly people." Health & Social Care in the Community 16.6 (2008): 593-605.
[2]   Battineni, Gopi, Nalini Chintalapudi, and Francesco Amenta. "AI chatbot design during an epidemic like the novel coronavirus." Healthcare. Vol. 8. No. 2. MDPI, 2020.
[3]   Roth, Dee, and Gerald J. Bean Jr. "New perspectives on homelessness: Findings from a statewide epidemiological study." Psychiatric Services 37.7 (1986): 712-719.
[4]   Cavallo, Eduardo, and Ilan Noy. "Natural disasters and the economy—a survey." International Review of Environmental and Resource Economics 5.1 (2011): 63-102.
[5]   Himmelstein, Jerome L., and James A. McRae Jr. "Social issues and socioeconomic status." Public Opinion Quarterly 52.4 (1988): 492-512.
[6]   Herman, Daniel, et al. "Critical Time Intervention: an empirically supported model for preventing homelessness in high risk groups." The Journal of Primary Prevention 28 (2007): 295-312.
[7]   He, Danping, et al. "Stochastic channel modeling for kiosk applications in the terahertz band." IEEE Transactions on Terahertz Science and Technology 7.5 (2017): 502-513.

[8]   Holfelder, Wieland, and Dietmar Hebmann. "A Networked Multimedia Retrieval Management System for Distributed Kiosk Applications." ICMCS. 1994.

[9]   Haristiani, Nuria. "Artificial Intelligence (AI) chatbot as language learning medium: An inquiry." Journal of Physics: Conference Series. Vol. 1387. No. 1. IOP Publishing, 2019.

[10]   High, Rob. "The era of cognitive systems: An inside look at IBM Watson and how it works." IBM Corporation, Redbooks 1 (2012): 16.

[11]   Moroney, Laurence, and Laurence Moroney. "The firebase realtime database." The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform (2017): 51-71.

[12]   Khurshid, Anjum, and Ashish Gadnis. "Using blockchain to create transaction identity for persons experiencing homelessness in America: policy proposal." JMIR research protocols 8.3 (2019): e10654.

[13]   Philipps, Kaitlin. "Homelessness: Causes, culture and community development as a solution." (2012).

[14]   Culhane, Dennis, et al. "The emerging crisis of aged homelessness in the US: could cost avoidance in health care fund housing solutions?." International Journal of Population Data Science 4.3 (2019).

[15]   Firat, Mehmet. "How chat GPT can transform autodidactic experiences and open education." Department of Distance Education, Open Education Faculty, Anadolu Unive (2023).